REMARKS

In the May 14, 2004, Office Action in this application, the United States Patent and Trademark Office (hereinafter the "Office") rejected Claims 12-31 under 35 U.S.C. § 102(e) as being anticipated in view of the teachings of U.S. Patent Application Publication No. 2002/0069189, filed by Bertrand et al. (hereinafter "Bertrand et al."). Claim 12 was also rejected under 35 U.S.C. § 101 because the Office alleged that the claimed invention is directed to non-statutory subject matter. Claim 12 was further rejected under 35 U.S.C. § 112, first paragraph, because the Office alleged that a rejection under 35 U.S.C. § 101 would require a corresponding rejection under 35 U.S.C. § 112, first paragraph. Claim 17 was rejected under 35 U.S.C. § 112, second paragraph, as being indefinite due to the claim limitations "other software" and "other facts." Claim 12 has been canceled without prejudice and disclaimer and Claim 17 has been amended, thereby rendering the rejections of those claims moot. Moreover, although Office has objected to the specification, the applicant has conformed to the objection thereby rendering the objection moot.

Prior to discussing in detail why applicant believes that all of the claims in this application are allowable, a brief description of applicant's invention and a brief description of the teachings of the cited and applied references are provided. The following background and the discussions of the disclosed embodiments of applicant's invention and the teachings in the cited and applied references are not provided to define the scope or interpretation of any of the claims of this application. Instead, such discussions are provided to help the Office better appreciate important claim distinctions discussed thereafter.

Background of the Invention

A user interface is a portion of a program or an operating system through which a user can instruct a computing device to accomplish a result and through which the device can convey

MSFT\18568AM1.DOC

information to the user. User interfaces can be constructed directly in the programming languages used by software programmers, but are more often constructed using specialized user interface development tools. For example, graphical user interfaces are often constructed using a tool called a forms package. A forms package typically presents the programmer with a screen (also called a form) that approximates what the user will see. The forms package allows the programmer to add individual graphical user interface controls (e.g., buttons, text entry boxes, list boxes) to the screen, and arrange the controls on the screen.

A graphical user interface for a program may consist of one or many screens. Forms packages allow the programmer complete freedom in constructing user interfaces with whatever screens the programmer desires. However, with this freedom comes the opportunity to make many mistakes. The programmer may create a user interface that is too complex for its users to understand and use properly. The programmer may have inadvertently created a user interface with bugs. An example of a bug is failing to handle correctly the entire range of possible input conditions by users. To reduce the likelihood of problems, programmers typically have learned to manually follow user interface guidelines in *de facto* conventions that suggest how user interfaces should appear and behave. As an example of the conventions, consider the standard placement of the OK button and the CANCEL button, which typically appear beneath other user interface controls. This convention stems from the fact that a user will interact with other user interface controls first and either the OK button or the CANCEL button second, and that people generally read a screen from top to bottom. It is unacceptable, therefore, to put the OK button or the CANCEL button at the top of the screen above other user interface controls, because a user would be likely to press the OK button or the CANCEL button before selecting an appropriate user interface option.

Portentously, the decision as to which control should be used is left entirely to the programmer. The programmer must evaluate the situation at hand, compare it to the available user interface guidelines, if any, and conventions, and then make an appropriate selection. Failure to select the appropriate user interface pattern may risk confusing users. Complicating the programmer's decision is that, at the time the programmer is writing the program, the programmer is typically unable to know the exact conditions under which the user interface will be used. A program may need to offer the user a list of choices where the number of choices varies greatly depending upon factors that change (e.g., the program needs to display a list of people currently connected to a computer network). The programmer is often forced to make bad decisions based on a theoretical or estimated range of values for such a factor. The decision made at the time of writing the program may result in a user interface that is inappropriate in practice.

Summary of the Invention

Applicant's invention is directed to a computer-based system, method, and computer-readable medium for moving much of the burden of identifying and constructing an appropriate user interface pattern to an expert system, which is programmed to follow guidelines, conventions, and principles of user interface design. A programmer writes an application in a traditional manner, but does not need to create a complete user interface for the application. Instead, the programmer writes code to reflect his intentions for the purpose of the user interface and these pieces of code invoke the expert system, which completes the user interface of the application. The expert system generates an appropriate user interface on the fly and returns this interface to the application. The application then invokes the user interface, which controls user interaction. The user interface communicates with the application as necessary. The user interface eventually returns control to the application when the user interface receives some

MSFT\18568AM1.DOC

indication from the user's interaction. Instead of generating the user interface on the fly, alternatively, the expert system generates and stores the user interface for later use during run time of the application. The programmer can use the expert system to generate an application's entire user interface or just a portion of it.

Summary of Bertrand et al.

Similar to various embodiments of applicant's invention, the system of Bertrand et al. uses an expert system, but the similarity ends there. The system of Bertrand et al. completely lacks the generation of user interface instructions by the expert system. The system of Bertrand et al. focuses on a goal-based educational system with personalized coaching. See the Title of the Invention of Bertrand et al. More specifically, the system of Bertrand et al. is directed to an educational tutorial system that analyzes student inputs to determine appropriate feedback to teach new skills. See the Field of the Invention of Bertrand et al. The feedback of Bertrand et al. includes text feedback, multimedia feedback, and reference material feedback. The system of Bertrand et al. uses feedback to correct the faulty business habits of students and raise the students' general business competence. There can be no reasonable interpretation that feedback of Bertrand et al. is somehow identical to applicant's generation of user interfaces by the expert system.

The principal concerns of Bertrand et al. are that present educational expert systems often suffer from a lack of motivational aspects that result in a student becoming bored or ceasing to complete a training program. Current training programs use static, hard-coded feedback with some linear video and graphics, which are used to add visual appeal and illustrate concepts. These educational systems typically support one correct answer and navigation through the educational systems is only supported through a single defined path resulting in a two-dimensional generic interaction. No business model support can be provided and only a

-12-

MSFT\18568AM1.DOC

single instance of feedback is provided to the student regarding whether a selected response is correct or not correct. The essence of the teachings of Bertrand et al. is that current educational tutorial systems do not architect real business simulations into the rules, which are used by the expert system, to provide a creative learning environment to a user. Bertrand et al. has nothing to do with the generation of user interfaces as in various embodiments of applicant's invention.

To solve these perceived problems associated with current educational tutorial systems, Bertrand et al. provides a goal-based learning system, which uses a rule-based expert training system to provide a cognitive educational experience. It is truly a mystery what this has to do with the inventive subject matter of applicant's invention, which has to do with generating user interfaces so that a user may use to interact with a computer system. The system of Bertrand et al. provides the user with a simulated environment that presents a business opportunity to understand and solve optimally. Mistakes are noted by the system of Bertrand et al. and remedial educational material is presented dynamically to build the necessary skills that a user requires for success in the business endeavor. In other words, the system of Bertrand et al. acts as a simulator to simulate a business environment in which a user is free to experiment to gain skills without real world financial repercussions. A robust business model is allegedly provided by the system of Bertrand et al. for supporting realistic activities and allowing users to experience real world consequences for their actions and decisions in a tutorial system, which analyses student inputs and determines appropriate feedback to teach new business skills.

The Claims Distinguished

The Office has failed to show, and applicant is unable to find, where any of the cited and applied references, either alone or in combination, disclose the subject matter of the claimed invention. For example, none of the applied and cited references teaches "in response to receiving said at least one intention and its associated set of parameters, the expert

-13-

LAW OFFICES OF
CHRISTENSEN O'CONNOR JOHNSON KINDNESS™
1420 Fifth Avenue
Suite 2800
Seattle, Washington 98101
206.682.8100

MSFT\18568AM1.DOC

system...generating user interface instructions from a template associated with the selected rule" as recited in Claims 13 and 31, among other claimed limitations. The system of Bertrand et al. uses an expert system to provide the user with a simulated business environment—but not for the purpose of generating user interface instructions from a template associated with a selected rule.

As explained in the background of the pending application, software programmers construct user interfaces using programming languages or specialized user interface development tools. The freedom to construct user interfaces comes with the opportunity to make mistakes, resulting in user interfaces that are too complex for their users to understand and use properly. The lack of knowledge by the programmer regarding the exact conditions under which the user interface will be used may result in a user interface that is inappropriate in practice. The system of Bertrand et al. suffers from these same problems. Bertrand et al. explains that "once the simulation model, system dynamics model and feedback are completely tested by designers, developers can incorporate [simulated business tasks] in a graphical user interface, e.g., Visual Basic, as a development platform." See paragraph 0884 of Bertrand et al. What this shows is that the system of Bertrand et al. manually creates user interfaces with programming languages or specialized user interface development tools, which is the traditional practice of creating user interfaces. The system of Bertrand et al. is not using its expert system to generate a user interface because its user interfaces are already generated using Visual Basic. The system of Bertrand et al. does not need to generate user interface instructions from a template associated with a selected rule using the expert system. Thus, the Office has failed to state a *prima facie* case of anticipation.

The Office has alleged that the claimed limitation "the expert system ... generating user interface instructions in accordance with the template associated with the selected rule" can be found at paragraph 0326 of Bertrand et al. There is nothing whatsoever at the cited paragraph

that can be reasonably interpreted to disclose the claimed limitation "the expert system ... generating user interface instructions from a template associated with the selected rule." To make this explanation clear, applicant recites paragraph 0326 of Bertrand et al. in full:

> The Intelligent Coaching Agent Tool (ICAT) is a suite of tools—a database and a Dynamic Link Library (DLL) run-time engine—used by designers to create and execute just-in-time feedback of Goal Based training. Designers write feedback and rules in the development tools. Once the feedback is set the run-time engine monitors user actions, files rules and composes feedback which describes the business deliverable. (Emphasis provided)

It would appear that the cited portion of Bertrand et al. by the Office teaches exactly opposite from the claimed limitation "the expert system ... generating user interface instructions from a template associated with a selected rule." The reason for that is because designers have to manually write feedback and rules in the development tools. The claimed limitation specifies that it is the expert system that generates user interface instructions. That cannot be found in Bertrand et al.

Among other differences, none of the applied and cited references teaches "an application including an incomplete user interface..., the incomplete user interface of the application being completed when the one or more intentions are realized" as recited in independent Claim 14, among other claimed limitations. The Office has argued that the system of Bertrand et al. teaches this feature of the claimed invention at paragraphs 0324, 0326, 0389, and 0234. This cannot be correct. Paragraphs 0324, 0326 of Bertrand et al. describe the use of development tools by designers to write feedback and rules. There is no mentioning of an application that

y

 Iняю

·includes an incomplete user interface, which becomes completed when one or more intentions are realized. Paragraph 0389 of Bertrand et al. discloses the following:

> When identifying problems, the tutor needs to prompt the student to reflect on a problem and start to point the student towards the answer. The tutor should not tell the student the answer, but instead should attempt to provide an appropriate answer or give the student a question to think about.

It is unclear what this has to do with Claim 14, which recites "a system for generating user interfaces so that a user may interact with a computer system." Paragraph 0234 of Bertrand et al. similarly fails to disclose anything that is even remotely relevant.

The Office has also failed to show, and applicant is unable to find, where the applied and cited references teach "generating a user interface by selecting a code module..., the act of selecting a code module including selecting a rule from a set of rules extracted from guidelines, conventions, and principles of user interface design..." as recited in Claims 19 and 25 among other limitations. The Office has again referred to paragraphs 0083, 0324, and 0326, which contain nothing whatsoever that disclose the claimed invention. Perhaps the confusion lies in the creation and delivery of feedback in the system of Bertrand et al.

Recall that the goal of Bertrand et al. is to produce an educational tutorial system that analyzes student inputs to determine appropriate feedback to teach new skills. For example, in paragraph 0322, it describes the use of a toolbar by a student to navigate to access features of a business simulation application including feedback. The student can have his business deliverables analyzed and receive feedback by clicking on a TEAM button. Thus, feedback is used to help a student in a business simulation. Feedback is content which may be presented via

a user interface to a user—but feedback as described by Bertrand et al. is not a user interface. The system of Bertrand et al. principally is concerned with the creation and delivery of feedback but not the generation of a user interface by an expert system. To provide further emphasis, the system of Bertrand et al. did not use the expert system to produce the toolbar with which the student navigates at paragraph 0322. Even the TEAM button is pre-generated by the system of Bertrand et al. The confusion regarding feedback as used by the system of Bertrand et al. must be dispelled to understand applicant's invention.

Because the Office has failed to state a *prima facie* case of anticipation, the rejection should be withdrawn. Independent Claims 13, 14, 19, 25, and 31 are clearly patentably distinguishable over the cited and applied references. Claims 15-18, 20-24, and 26-30 are allowable because they depend from allowable independent claims and because of the additional limitations added by those claims. Consequently, reconsideration and allowance of Claims 13, 14-31 is respectfully requested.

CONCLUSION

In view of the foregoing remarks, applicant submits that all of the claims in the present application are clearly patentably distinguishable over the teachings of Bertrand et al. Thus, applicant submits that this application is in condition for allowance. Reconsideration and reexamination of the application, allowance of the claims, and passing of the application to issue at an early date are solicited. If the Examiner has any remaining questions concerning this

MSFT\18568AM1.DOC

application, the Examiner is invited to contact the applicant's undersigned attorney at the number below.

Respectfully submitted,

CHRISTENSEN O'CONNOR
JOHNSON KINDNESS^PLLC

D.C. Peter Chu
Registration No. 41,676
Direct Dial No. 206.695.1636

CERTIFICATE OF MAILING

I hereby certify that this correspondence is being deposited with the U.S. Postal Service in a sealed envelope as first class mail with postage thereon fully prepaid and addressed to Mail Stop Amendment, Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450, on the below date.

Date: _August 16, 2004_   _Cindy A. Morton_

DPC:ejh

MSFT\18568AM1.DOC